REMARKS

These remarks are in response to the Office Action dated June 19, 2002. Claims 1-28 are pending in the present application.

Present Invention

A system and method in accordance with the present invention is described in the Summary of the Invention, page 3, line 16 to page, 4, line 13, which is reproduced in its entirety herein below.

> **A system and method for providing a command stream on a chip or in a computer system is disclosed. The system comprises a central processing unit (CPU), a controller coupled to the CPU and a memory coupled to the controller. The controller manages the memory. The system and method also includes a storage element coupled to the memory, the storage element being accessible by the CPU via the controller.**
>
> **The system and method in accordance with the present invention allows the controller to receive commands from the CPU and to manage the storage element, typically a first in first out (FIFO) buffer, and incorporates the storage element as part of the memory. In so doing, the system performance is significantly improved by providing a virtual FIFO buffer such that the CPU sees a FIFO buffer with a size equal to the size of the memory. The bandwidth of the bus between the controller and the memory is typically greater than that of the system bus. Hence, the performance of the overall system is significantly improved. In addition, since the controller is the only device that has access to the memory, the bandwidth of the bus associated therewith is not divided between two devices. Further, since the controller is managing the storage element, the overhead considerations related to managing the read and write pointers of the storage element by the CPU are eliminated. Finally, a technique in accordance with the present invention is provided such that the controller determines whether the storage element must be emptied, rather than requiring the CPU to perform this function. Through these features, a command stream can be provided efficiently on a chip or in a computer system.**

Claim Rejections – 35 USC § 102

The Examiner states:

4. **Claims 1, 2, 9-12, 16-17, 20-21 and 25-26 are rejected under 35 U.S.C. 102 (e) as being anticipated by Dye, US Patent no. 6,173,381.**

5. **In re claim 1, Dye shows a controller chip [fig 5] comprising: an engine [210, 212, fig 5] for managing a memory [204, 206, 214, 216, 244, fig 5] and including an interface [202]; and a storage element [204, 206, 216, 244, 246, fig 5] coupled to the engine, the storage element being accessible by a CPU [120, fig 2], where the engine receives commands from the CPU via the interface, manages the storage element and write the commands into the memory.**

6. **In re claim 2, Dye shows a FIFO buffer [204, 206, 214, 216 fig.5].**

7. **In re claim 9, Dye shows a graphics controllers chip [212, fig 5].**

8. **In re claim 10, Dye shows a graphics engine [212, fig 5].**

9. **In re claim 17, Dye shows the effective size of the FIFO buffer as viewed by the CPU can be as large as the memory [fig 5].**

10. **Claims 11-12, 16, 20-21, 25-26 are rejected under the same rationale as discussed above in claims 1, 2, 9, 10, 17.**

Applicant respectfully disagrees.

For ease of review, independent claims 1, 11 and 20 are reproduced in their entirety herein below:

1. **A controller chip comprising:**

an engine for managing a memory and including an interface; and

a storage element coupled to the engine, the storage element being accessible by a central processing unit (CPU), wherein the engine receives commands from the CPU via the interface, manages the storage element and writes the commands into the memory.

11. **A system for providing a command stream in a computer system comprising:**

a central processing unit (CPU);

a controller coupled to the CPU and including an interface;

a memory coupled to the controller, the memory being managed by the controller; and

a storage element coupled to the controller, the storage element being accessible by the CPU, wherein the controller receives commands from the CPU via the interface, manages the storage element and writes the commands into the memory.

20. A method for providing a command stream in a computer system, the computer system including a central processing unit (CPU), a controller coupled to the CPU, a memory coupled to the controller, the memory being managed by the controller; the method comprising the steps of:

(a) providing a storage element within the controller; and

(b) allowing the storage element to be accessible by the CPU via an interface in the graphics controller.

Applicant submits that Dye neither teaches nor suggests the invention as recited in the above-identified independent claims. Dye discloses an integrated memory controller 140 which Examiner has indicated discloses the features of the present invention. However, Dye is similar to the prior art disclosed within the present application and has problems similar thereto. Dye discloses, at column 12, line 61 through column 13, line 50:

> **As shown, the IMC 140 includes bus interface logic 202 for coupling to the host computer system, i.e., for coupling to the system bus 106. In the preferred embodiment, the system bus 106 is the CPU bus or host bus. Alternatively, the system bus 106 is the PCI bus, and the bus interface logic 202 couples to the PCI bus. Instruction storage/decode logic 230 is coupled to the bus interface logic 202.**
>
> **The bus interface logic 202 couples to an execution engine 210 through two first in first out 9FIFO) buffers 204 and 206. In other words, the two FIFO buffers 204 and 206 are**

coupled between the bus interface logic 202 and the execution engine 210. The FIFO buffers 204 and 206 decouple data transfers between the external asynchronous computer system and the synchronous logic comprised within the IMC 140. The execution engine 210 includes a data compression/decompression (codec) engine according to the present invention, as described further below. The execution engine 210 also include (Sic.) texturing mapping logic for performing texture mapping on pixel data. In one embodiment, the execution engine 210 includes separate compression and decompression engines.

The execution engine 210 couples to a graphics engine 212. The graphics engine 212 essentially serves as the graphical adapter or graphics processor and includes various graphical control logic for manipulating graphical pixel data and rendering objects. The graphics engine 212 includes polygon rendering logic for drawing lines, triangles, etc., i.e., for interpolating objects on the display screen 142. The graphics engine 212 also includes other graphical logic, including ASCII to font conversion logic, among others. The instruction storage/decode logic 230 stores instructions for execution by the graphics engine 212.

In one embodiment, the execution engine 210 comprises a DSP engine which performs both codec functions as well as graphical functions. In one embodiment, the DSP engine includes one or more ROMs which store different microcode depending on the task being performed, and the DSP engine dynamically switches between different sets of microcode to perform different tasks.

The graphics engine 212 couples to respective memory control units referred to as memory control unit #1 220 and memory control unit #2 222 via respective FIFO buffers 214 and 216, respectively. Memory control unit #1 220 and memory control #2 222 provide interface signals to communicate with respective banks of system memory 110. In an alternate embodiment, the IMC 140 includes a single memory control unit. The graphics engine 212 reads graphical data from system memory 110, performs various graphical operations on the data, such as formatting the data to the correct x, y addressing, and writes the data back to system memory 110. The graphics engine 212 performs operations on data in the system memory 110 under CPU control using the high level graphical protocol. In many instances, the graphics engine 212 manipulates or resets pointers and manipulates data in windows workspace areas in system memory 110, rather than transferring the pixel data to a new location in system memory 110.

Referring to this passage in conjunction with Figure 5 of Dye (copy attached), it can be readily seen that there are significant differences between the recited invention and the Dye

reference. They will be described with particularity herein below.

1. The claimed controller chip comprises an engine for managing a memory and including an interface.

Dye discloses at column 13, lines 1-5:

> **The bus interface logic 202 couples to an execution engine 210 through two first in first out (FIFO) buffers 204 and 206. In other words, the two FIFO buffers 204 and 206 are coupled between the bus interface logic 202 and the execution engine 210.** (emphasis added)

Accordingly, it is clear from this passage that the interface in Dye is between the execution and the interface. As stated above, in the recited invention the engine includes an interface. The interface being included in the engine is very important because, as described in the specification at page 8, line 17 through page 9, line 6 and as seen in Figure 4 of this present application:

> **The graphics engine 302 accepts a command stream through an interface 303 from the CPU 12." The graphics engine 302 in turn can utilize the graphics memory 26" as well as a storage element, such as a FIFO buffer 306, as the buffer for CPU purposes. Accordingly, the effective size of the entire buffer can be virtually the size of the graphics memory 26". Therefore, the CPU 12" will need to check whether the FIFO buffer 306 is full much less frequently than in the above conventional systems since the effective size of the FIFO buffer is much larger.** (emphasis added)

and page 9, lines 5-6:

> **Accordingly, from the CPU's point of view, the interface 303 is the storage element. In fact, it is a storage element with a size equal to the size of graphics memory 26".**

Accordingly, by including the interface within the engine rather than having it separated.

therefrom by FIFO elements, the overall performance of the system is improved.

2. In the claimed controller the engine receives commands from the CPU via the interface, manages the storage element and writes the commands into the memory.

The engines in Dye (execution engine and graphics engine) do not manage the FIFOs. As disclosed in Dye, at column 13, lines 5-8:

> **The FIFO buffers 204 and 206 decouple data transfers between the external asynchronous computer system and the synchronous logic comprises within the IMC 140.**

and at column 13, lines 33-38:

> **The graphics engine 212 couples to respective memory control units referred to as memory control unit #1 220 and memory control unit #2 222 via respective FIFO buffers 214 and 216, respectively. Memory control unit #1 220 and memory control #2 222 provide interface signals to communicate with respective banks of system memory 110.**

Accordingly, as is seen, the storage elements (FIFOs) are not managed by the engine in Dye, but are utilized as decoupling points for data transfers. This system is more similar to that of Figure 2 (copy attached) in the present application. Referring now to the application at page 6, line 13 through line 20:

> **In this system 100 (Figure 2), the CPU 12' directly accesses a FIFO buffer 106 within the graphics controller 24' and the FIFO buffer 106 provides that data to a graphics engine 102. Typically, the FIFO buffer 106 is a circular FIFO buffer. As has also been above-explained, the CPU will have to either poll the graphics engine 102 to ensure that the FIFO buffer 106 is not full, or the graphics engine 102 will interrupt the CPU periodically to ensure that the FIFO buffer 106 is not full. As has been above mentioned, both of these alternatives adversely affect system performance, and more particularly, in multiprocessing environments polling is the only option.**

Accordingly, as is seen, the system of Dye would have the same deficiency as disclosed in

relation to Figure 2, that is, the CPU would need to be interrupted periodically to poll the FIFOs as to their status. As before mentioned, by placing the interface within the engine, the interface from the CPU's perspective is the storage element, and the resultant storage element can be as large as the memory. Therefore, polling by the CPU is significantly reduced.

Accordingly, Applicant submits that the present invention as recited in independent claims 1, 11 and 20 is neither taught nor suggested by the cited reference. For the above-stated reasons, Applicant respectfully requests reconsideration and allowance of the claims as now presented. In addition, claims 2, 9, 10, 12, 16, 17, 21, 25 and 26 are allowable because they depend from an allowable base claim.

Claim Rejections – 35 USC § 103

The Examiner states:

> **12. Claims 3-8, 13-15, 18019, 22-24, 27, and 28 rejected under 35 U.S.C. 103(a) as being unpatentable over Dye, US Patent no. 6,173,381, as applied to claims 1, 2, 9, 10-12, 16-17, 20-21 and 25-26 above.**
>
> **13. In re claims 3-5 and 7-8, Dye does not explicitly show a circular FIFO buffer, a double buffer, a triple buffer, a checking mechanism. Official Notice is taken that both the concept and the advantages of providing for a circular FIFO buffer, a double buffer, a triple buffer, a checking mechanism are old and well known in the art. Therefore, it would have been obvious to the ordinary skilled person in the art at the time the invention was made to include the circular FIFO buffer, double buffer, triple buffer, checking mechanism in Dye for more flexible device by allowing it to operate in multiple configurations and more reliable system by controlling and predicting data flow.**
>
> **14. In re claim 6, Dye shows the effective size of the FIFO buffer as viewed by the CPU can be as large as the memory [fig 5].**
>
> **15. Claims 13-15, 18-19, 22-24, 27, and 28 are rejected under the same rationale as discussed above in claims 3-8.**

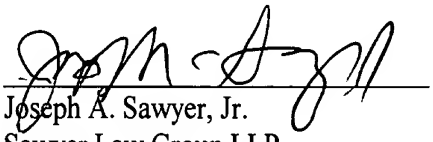
The above-identified claims are also allowable for at least the reasons stated with respect to the above-identified, allowable claims. In addition, with respect to claims 7 and 8, there is no

teaching or suggestion of a checking mechanism for determining if the FIFO buffer needs to be emptied without utilizing the CPU, as recited in claim 7. Furthermore, there is no teaching of the elements of checking mechanism recited in claim 8. Applicant submits that Examiner must supply a reference for this rejection and that taking Official Notice that a concept is old or well-known is not sufficient for a rejection of the dependent claims. These arguments also apply to claims 18, 19, 27 and 28.

In view of the foregoing, Applicant submits that claims 1-28 are allowable. Applicant, therefore, respectfully requests reconsideration and allowance of the claims as now presented.

Applicants' attorney believes this application in condition for allowance. Should any unresolved issues remain, Examiner is invited to call Applicants' attorney at the telephone number indicated below.

Respectfully submitted,



Joseph A. Sawyer, Jr.
Sawyer Law Group LLP
Attorney for Applicant(s)
Reg. No. 30,801
(650) 493-4540

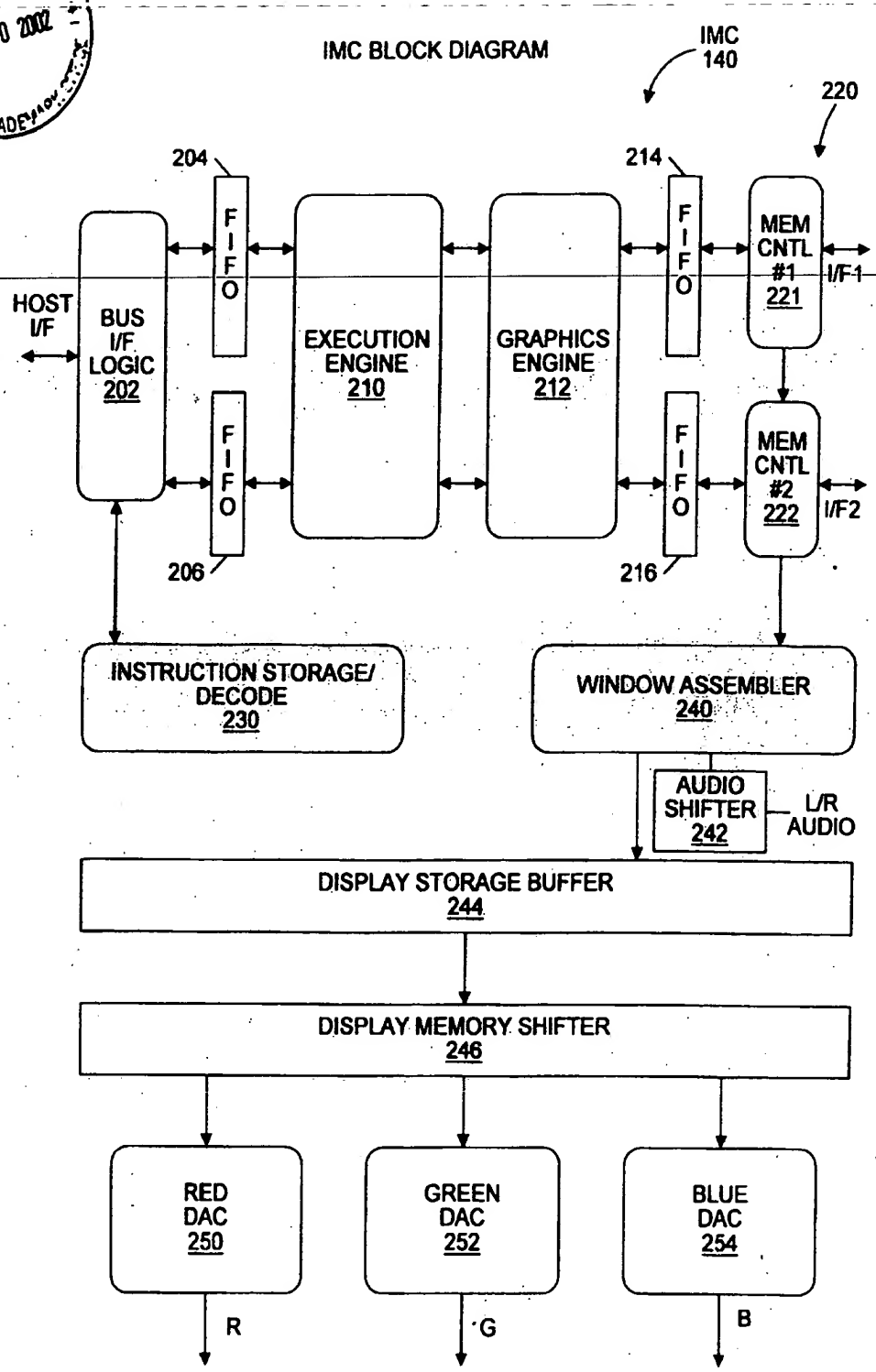


FIG. 5

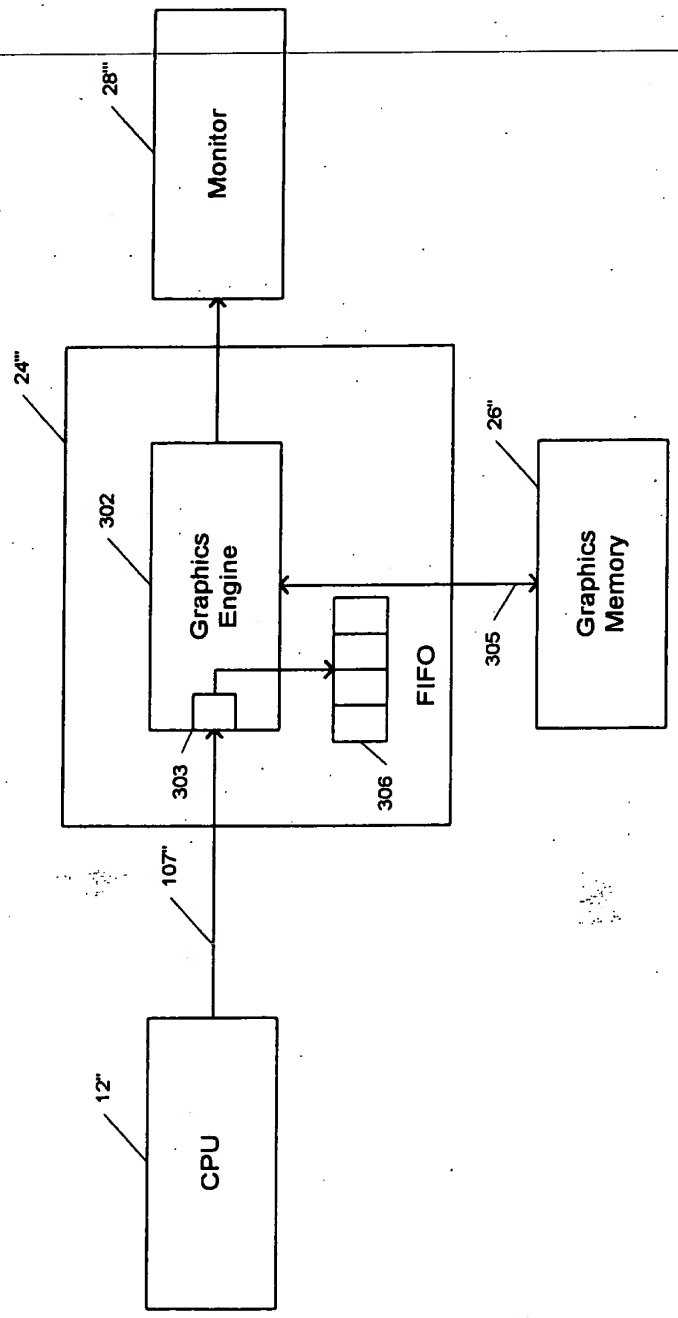
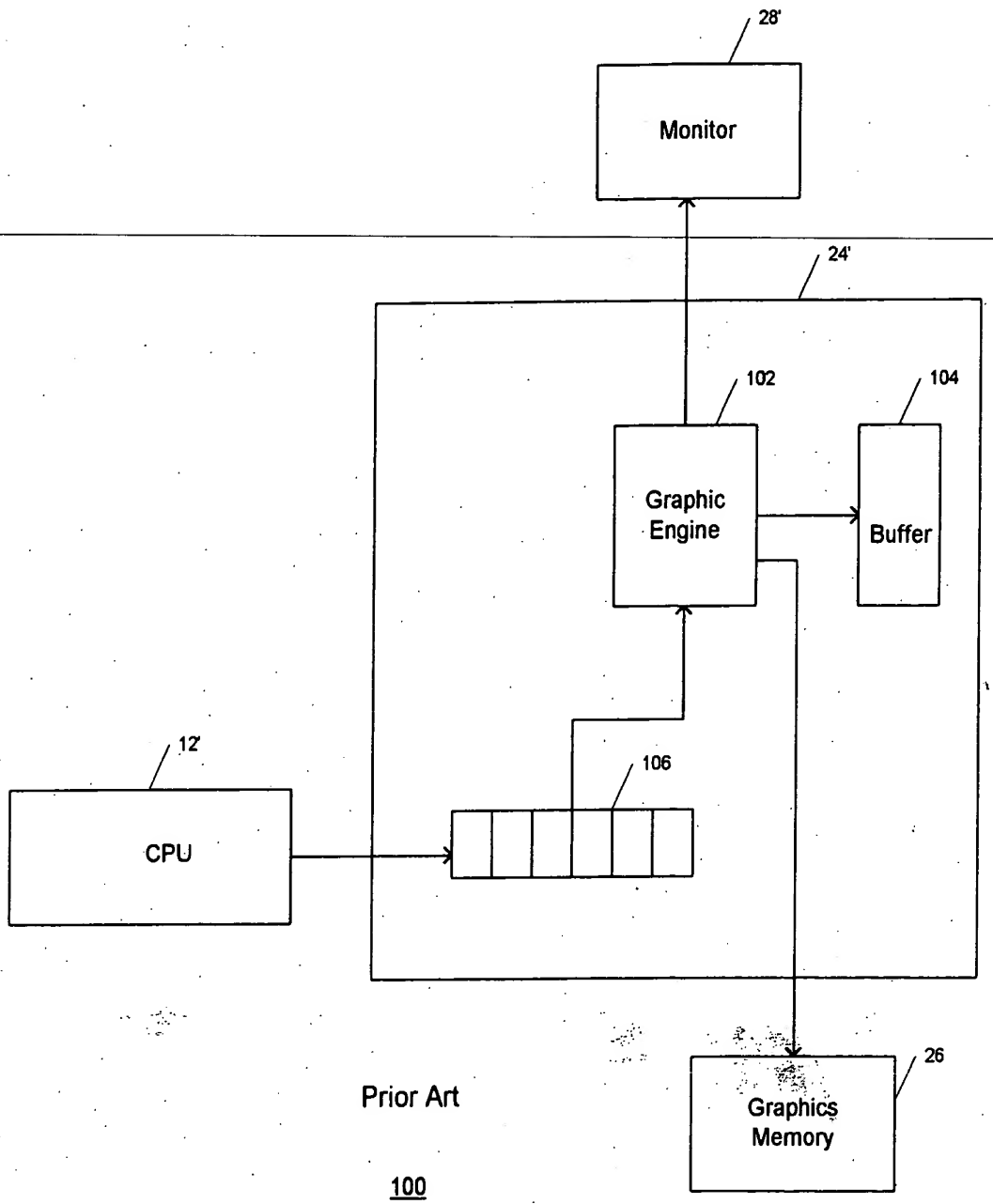


Fig. 4



Prior Art

100

Fig. 2